Volume 6
February 2016

NEWSLETTER

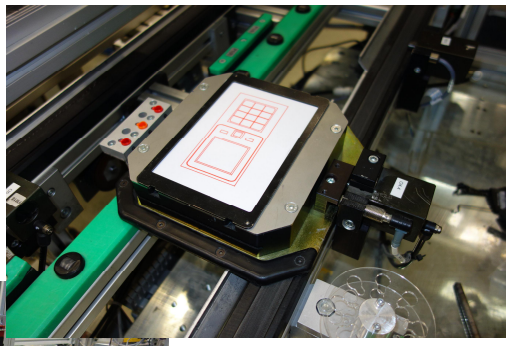
Welcome

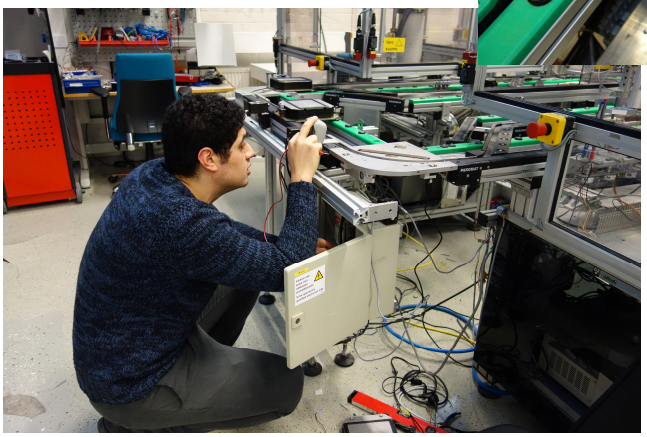
Welcome to the sixth edition of the eScop Newsletter!

The eScop project has proceeded to its final month. In this extra-large edition of the eScop newsletter we are sharing the latest news and achievements from the project.

During late autumn and winter the Consortium has kept busy doing the final technical developments and finishing reporting all of the activities in deliverables. During the winter the Consortium created and mutually agreed on an agreement on the intellectual property rights. Technical workshops have been held online to both advance the eScop solution and to train consortium members internally.

Effort has been put to show all the achievements in the final technical review, taking place in Finland in February 2016. The review includes not only theoretical presentations but also practical visits to see the demonstrators in Fluidhouse in Jyväskylä and FASTory in Tampere.





Figures 1 and 2. Preparing FASTory line for final review meeting

The project has reached the two last milestones. The milestone 6, fully implemented pilots, was achieved in month January 2016 by finalization of WP7 deliverables related to the tasks on pilots. The seventh and last milestone was validation and usability assessment. It was represented in the deliverables Guideline on eScop methodology (D6.6), Best practices guideline (D7.5), eScop eBook (D8.7) as demonstration that eScop architecture is validated and is ready with the proper documentation.

The eScop consortium has continued to disseminate the results. TUT participated in the ICT 2015 event in October 2015 with a 45-minute networking session, sketching the future of the industry with like-minded individuals through a method called Six Thinking Hats.

The project has also been preparing training materials so that the eScop solution can easily be adopted outside the consortium. Extensive training materials will be available at the eScop website.

In order to sustain the project results and to affect future standards the Consortium also launched a W3C Community called the open Knowledge-driven Service-oriented System architectures and APIs (KiSS) community group and marketed it in several events.

Regarding the newsletter, we would especially like to draw your attention to the section about eScop results on page 2. In addition this edition presents the eScop remote terminal units and discusses the MES framework. Contact details and a dissemination story from 12th European Workshop on Advanced Control and Diagnosis can be found on the last page.

www.escop-project.eu

Project Details:

The **eScop**, "Embedded systems for Service-based control of Open Manufacturing and Process Automation", develops a core of the Manufacturing Execution System (MES) that overcomes current problems of system integration from shop-floor control to factory management levels by merging the power of knowledge representation, service orientation and embedded systems.

Duration
March 2013 – February 2016

Total Costs
5,82 M€

Participating Countries
Czech Republic, Finland, Italy and Poland

Participating Organisations
10

Inside this Issue

eScop at the finish line

Three years of the eScop project flew by and it is time to check how the eScop project looks at the finish line. The project has set an ambitious goal to develop a solution for Open Knowledge-Driven Manufacturing Execution System (OKD-MES). The project proposed the architecture utilizing web standards for integrating MES functions. It was interesting to behold how, first being quite general but setting up the direction, the architecture has been gradually refined with the details for the software and hardware components. The strong push for the web standards at the factory floor and, therefore, for MES solution is not an attempt to follow some current and 'trendy' technologies, it is an investment into sustainability of the solutions built following the eScop approach.

Figure 3. Raspberry PI (small grey box between keyboard and the screen) hosts the orchestrator and testers for the FASTory line. It is probably the cheapest item seen in the photo.

For example, the main tool one needs is a web browser. The web browser can be used to program industrial controller or to create a knowledge model or to start the orchestration of a process or to see the dynamically configured visualization screen... Having web browsers as the main platform for engineering tools increases the probability that the solutions developed to work with a web browser can be accessed still in decades to come due to scale and validation the standards went through.

The eScop project has proven that one can use simple, affordable devices that cost few tens of euros, e.g. Raspberry PI, for controlling a production line. Modern devices have sufficient computational and communication resources to run reliably different types of applications at different levels of the automation pyramid including factory floor, MES level and if necessary higher-level applications.

Figure 4 eScop book traveling through the knowledge-driven production line as the ideas and concepts outlined first in the paper were realized in the hardware.

The models created in Web Ontology Language (OWL), a standard by World Wide Web consortium (W3C), used at run time to support different layers staring from dynamic visualization to the device description. The project created the tools that can be used to check and refine the models (**www.escop-project.eu/tools**), which may not necessarily belong to the domain of industrial automation. Again, the user can use the web browser to upload, navigate and query the models. The tools web page mentioned above contains a list of tools and simulators that can be used to develop, refine and test the eScop solution. One can develop own RTU or the orchestration solution referring to the binaries accessible through the web site.

Each component went through thorough tests to evaluate its performance. How many subscribers can be there for the events coming from a device? What is the response time for the knowledge update? What is the performance of the orchestrator with respect to the amount of processes it has to handle? Those and similar basic questions are answered in the project deliverables to setup a firm base for developing reliable solutions to bring knowledge-driven approach into practical use.

The project has published the 'eScop book' (see photo above): https://www.researchgate.net/publication/282652628_Open_Knowledge_Driven_Manufacturing_and_Logistics_-_the_eScop_Approach, outlining the concepts, many of which were realized in the eScop solution for the OKD-MES. The project ambition goes beyond the formal duration of the project, as we have established a W3C community group for open Knowledge-driven Service-oriented System architectures and APIs (KiSS): https://www.w3.org/community/kiss/, where we intent continue learning, developing and disseminating ideas and solutions that were born thanks to the eScop project.

The project results were achieved thanks to the work of the consortium of the 10 project partners. In order to briefly highlight for the readers certain aspects, I would say that expertise on the eScop-related technologies can be asked from SCM Group and THT Control; POLIMI can be addressed for getting in particular details about eScop Manufacturing Systems Ontology, INCAS and Fluidhouse can provide end user perspective as well as eScop solutions integrated with their products; UWB is the strong expert for the solutions at the physical layer; ESPEO holds the expertise on the orchestration of service-oriented systems, while ICONICS can help defining and building dynamic user interfaces; WUT can put in the context the theoretical background to follow in order to develop knowledge-driven approach – if you would like to obtain the eScop book, please contact WUT; and, finally, TUT would be happy to assist with creation and deployment of the eScop ecosystem.
We would happy to hear the experience of other developers on using eScop components and tools.

eScopRTU - A universal control unit for OKD-MES

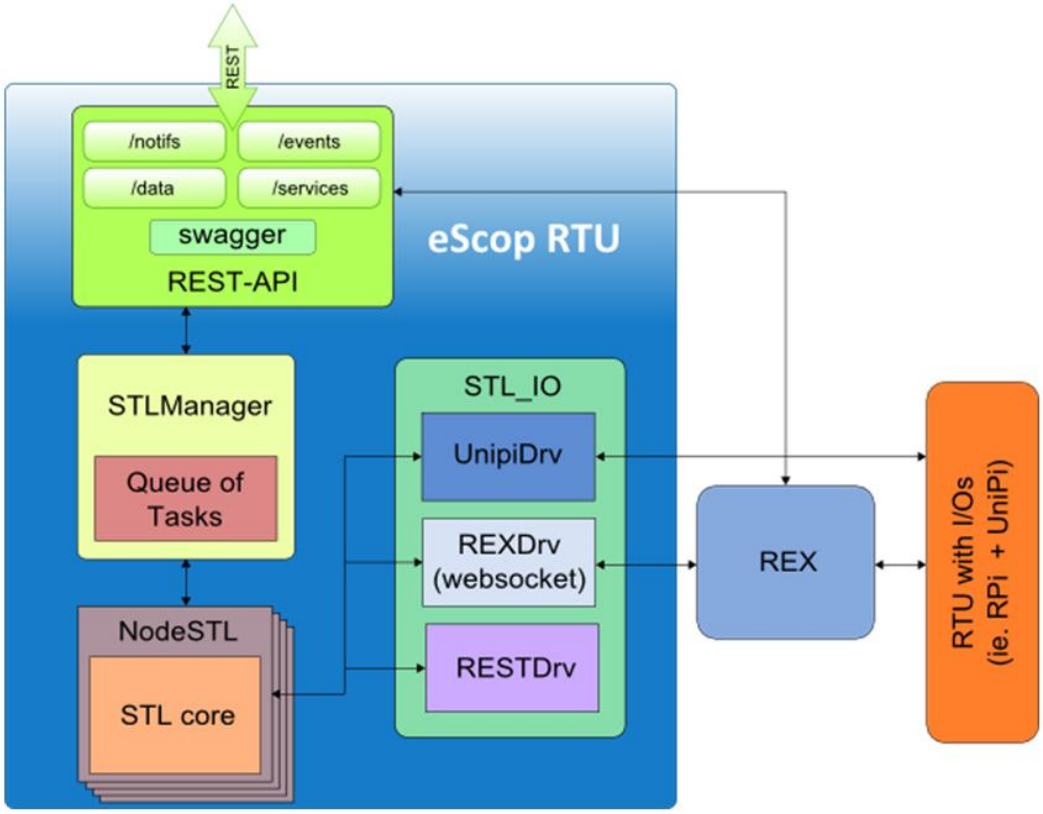


Figure 4. Schema of components of eScopRTU with STL service manager

An eScopRTU is a highly configurable PLC -like embedded device developed in Node.js environment for low level physical layer of Open Knowledge-Driven Manufacturing Execution System. One part of the device is a REST interface so communication with the device is very easy and cross-platform. It provides bridge between controlled process and service oriented architecture (SOA). The eScopRTU is composed from three main parts: REST web service interface, STL Manager and Device drivers. It supports discovery protocol, so each RTU proactively reports its IP addresses towards the other parts of the eScop system especially to the ontology service in representation layer.

There are two ways how to expose REST web services for other parts of the eScop system or outside world. First way is to configure eScopRTU with instructions in the Structured Text Language (STL) the second one uses JSON configuration file which exposes the services from the underlying real-time control system REX.

For a pleasant user experience the eScopRTU comes with a web-based integrated development environment (IDE). Using this application, a user can easily change unit configuration parameters such as device ID, port, etc. This user interface also contains a highly customized text editor that besides other functions features syntax highlighting and auto-completion, which makes it a perfect tool for STL scripts implementation. Downloading and management of these scripts can also be done directly from this interface. All this together with an interactive guide that is also embedded to the IDE should fulfill every user's needs.

The eScopRTU comes in precompiled version for Windows, Linux and Raspberry Pi microcomputer. It contains drivers for UniPi board extension, other REST services and finally for real-time control system. The whole application is distributed as an open-source under Apache License. It is available from eScop project webpage http://www.escop-project.eu/tools/eScopRTU

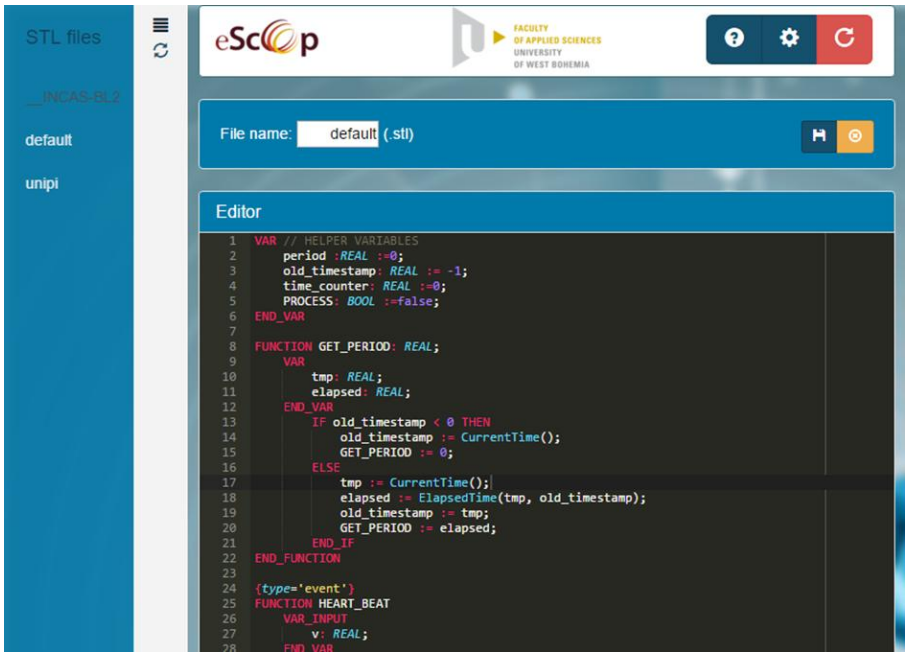


Figure 5. Web-based IDE

A framework to host MES

MES Framework, as the name suggests, is a skeleton on top of which MES functions can be built using a reusable set of libraries that compose the framework. MES Framework was developed as part of eScop project for constructing MES functions based on MESA standards. The framework operates along with eScop architecture to provide MES functionality, which is flexible and reconfigurable. Like the eScop RTUs, the framework also delivers MES functionalities as events and services, which can be accessed via RESTful web service based JSON messages. The framework and the MES functions

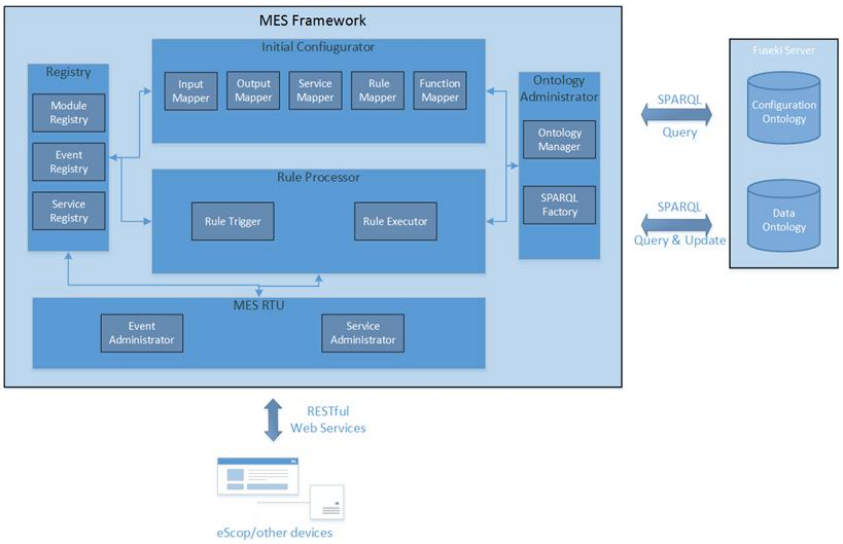


Figure 7. MES framework Architecture

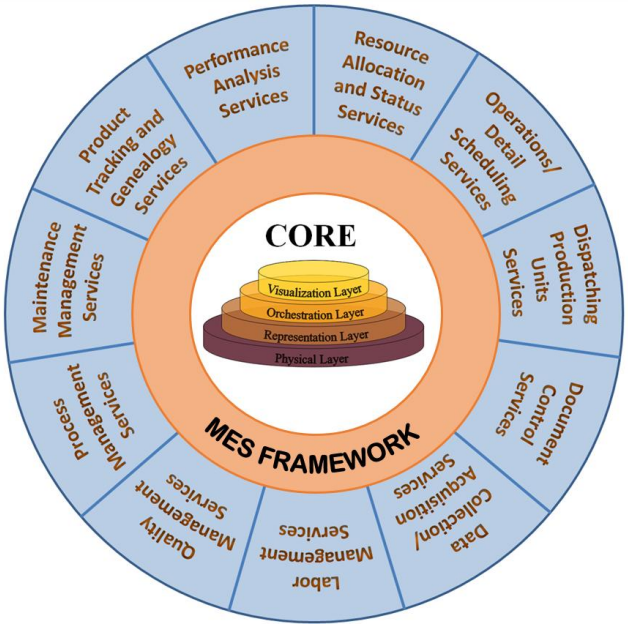


Figure 6. MES Framework in eScop architecture

surrounding the eScop architecture can be seen in figure 5.

Built on the OKD-MES concept, the framework is also knowledge driven. It uses two basic ontologies to hold the information: Configuration ontology holds the functionalities of MES; Data ontology holds the data generated while the functions are executing. The Framework builds itself in a series of steps. First, the user constructs the configuration ontology and hosts it in the server. Then the Framework builds the MES function based on the Configuration ontology and by interacting with RPL. Once the functions are built, it starts executing the functions. The architecture that helps in accomplishing the above tasks can be seen in figure 7.

Initially, the MES framework was planned with the view of hosting MESA based MES function and executing them. Later, once built it was realized that the MES framework while working with eScop Architecture can provide much more than just MES functions. It can hold functions to monitor devices; execute concepts like six-sigma, lean manufacturing, etc.; act as an business layer tool. In the eScop project, the MES framework was successfully tested with most of the above concepts.

Additional details about MES framework will be also published in the Thesis "A Methodology for the Development of Manufacturing and Monitoring Indexes: Oil Lubrication System Case Study" by Balaji Gopalakrishnan. The thesis is expected to be published and publicly accessible online from the Spring 2016. For the source files of the MES framework, please contact FAST-Lab (fast@tut.fi) and check for the updates at *http://www.escop-project.eu/tools.*

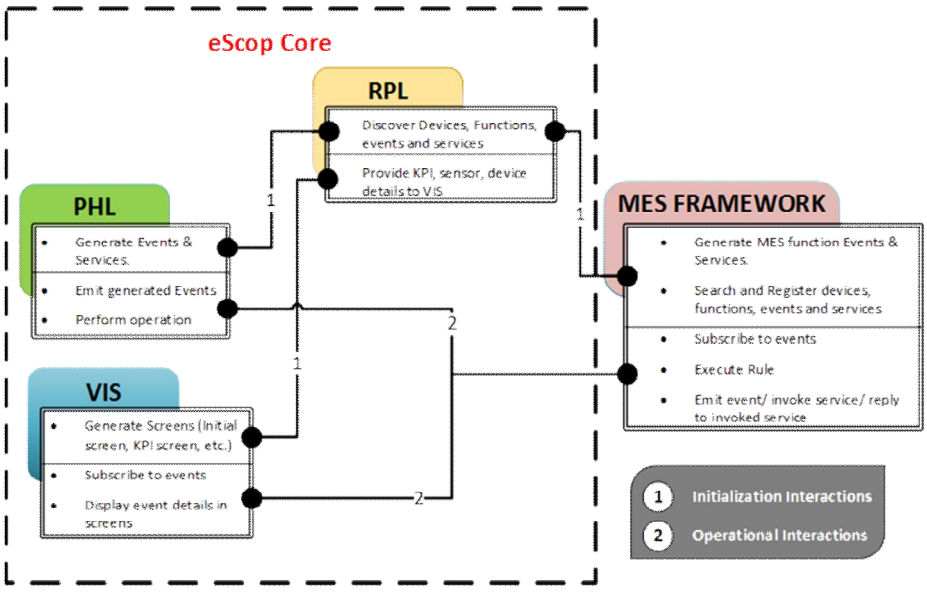


Figure 8. Interactions between eScop and MES framework

12th European Workshop on Advanced Control and Diagnosis

Figures 9-11. Participation on 12th European Workshop on Advanced Control and Diagnosis

The University of West Bohemia (specially the The European Centre of Excellence NTIS) hosted the the 12th European Workshop on Advanced Control and Diagnosis from 19-20th of November 2015. There were more that 70 participants from 18 different countries all around the world.

The discussion about results of eScop project and utilization of knowledge during control was opened as a part of the poster session. The current results of our project was introduces to professors from Coventry University, UK and Delft University of Technology, NL.

Knowledge-Driven Systems
From Enterprise Solutions to Embedded Devices

Contact

Coordinated by
Tampere University of Technology

Project Leader: Prof. Jose L. Martinez Lastra, jose.lastra@tut.fi
Project Manager: Johanna Rytkönen, johanna.k.rytkonen@tut.fi
Technical Coordinator: Dr. Andrei Lobov, andrei.lobov@tut.fi
www.escop-project.eu

eScop is made possible by funding from the ARTEMIS Joint Undertaking.